



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/837,205	04/19/2001	Shinichiro Eto	2001_0469A	1208

513 7590 03/17/2005

WENDEROTH, LIND & PONACK, L.L.P.  
2033 K STREET N. W.  
SUITE 800  
WASHINGTON, DC 20006-1021

EXAMINER

TRUONG, CAMQUY

ART UNIT	PAPER NUMBER
----------	--------------

2127

DATE MAILED: 03/17/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

**Office Action Summary**

Application No.

09/837,205

Applicant(s)

ETO ET AL.

Examiner

Camquy Truong

Art Unit

2127

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

- 1) ☒ Responsive to communication(s) filed on 11/9/04.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

- 4) ☒ Claim(s) 1-31 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 1-31 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

**Application Papers**

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on \_\_\_\_\_ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

**Priority under 35 U.S.C. § 119**

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some \* c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
  2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
  3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
- \* See the attached detailed Office action for a list of the certified copies not received.

**Attachment(s)**

- |  |   |
|--|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892)  | 4) <input type="checkbox"/> Interview Summary (PTO-413)<br>Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948)                                   | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152)             |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)<br>Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____  |

## DETAILED ACTION

1. Claims 1-31 are presented for examination.

### ***Claim Rejections - 35 USC § 103***

2. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

3. Claims 1-7 are rejected under 35 U.S.C. 103(a) as being unpatentable over Milot et al (U.S. patent 6,437,788 B1).
4. As to claim 1, Milot teaches the invention substantially as claimed including: a real-time OS simulator running on OS, said simulator being operable to respectively assign a plurality of threads to a plurality of task of real-time OS to be simulated, and to simulate an operation of the real-time OS on the OS (col. 4, lines 58-62), said simulator comprising:

A system function, called by one of the plurality of threads of the OS which is assigned as a first task processing thread, operable to communicate with another of the plurality of threads of the OS for providing an instruction thereto so as to switch a state of the currently running task of the simulated real-time OS from that of the first task processing thread to that of a thread of the OS which is

assigned as a second task processing thread (col. 4, lines 61-65; col. 5, lines 13-16; col. 6, lines 47-51); and

A task switching thread operable to receive from said system function the instruction to switch a state of the currently running task of the simulated real-time OS by resuming a plurality of threads of the OS corresponding to task processing threads (col. 5, lines 13-15 and lines 30-39).

5. Milot does not explicitly teach that the system is multi-thread OS. However, Milot teaches the Special Purpose Thread #1 and Thread #2 (Fig.4). It would have been obvious to one of ordinary skill in the art at the time the invention was made that in fact Milot's system including multi-thread as claimed by applicant, and having multi-thread OS would improve the method of managing graphics displays for computer systems.

6. As to claim 2, Milot teaches system function is operable to select a task processing thread to run next, to provide the instruction to said task switching thread for switching the currently running first task processing thread to the thread of the OS which is assigned as the selected task processing thread, and to then suspend the first task processing thread (col. 2, lines 42-45; col. 4, lines 58-61; col. 5, lines 13-18, and lines 30-35), and

Said task switching thread is operable to, in response to the instruction provided by said system function, resume the selected task processing thread

after the first task processing thread is suspended (col. 2, lines 52-60; col. 5, lines 30-35).

7. As to claims 3-4, Milot teaches system function is operable to select a task processing thread to run next, to provide the instruction to said task switching thread for switching the currently running task to the selected task processing thread (col. 6, lines 48-51), and to then set the thread of the OS corresponding to the first task processing thread that has issued the request in a waiting state (col. 5, lines 17-19), and

Said task switching thread is operable to, in response to the instruction, suspend the thread of the OS corresponding to the task (col. 5, lines 13-18), and then to release the selected task processing thread from a previously set waiting state for resuming (col. 5, lines 30-35).

8. As to claims 5, Milot teaches system function is operable to provide the instruction to said task switching thread after said task switching thread has been indicated as being enabled to start processing (col.5, lines 14-16).

9. As to claim 6, Milot teaches:

Said system function is operable to provide the instruction to said task switching thread after having selected the second task processing thread as the task processing thread to run next (col.6, lines 42-51), and

Said task switching thread is operable to run with a higher priority than said task processing threads and, in response to the instruction, to suspend a preceding running task processing thread and then resume the selected second task processing thread (col. 5, lines 14-18 and lines 30-35).

10. As to claim 7, Millot teaches task processing thread creating means for creating said task processing thread (col.4, lines 58-61).

11. Claims 14-17, 20-23 and 26 are rejected under 35 U.S.C. 103(a) as being unpatentable over Milot et al (U.S. patent 6,437,788 B1), as applied to claim 1 above, in view of Ranganathan (U.S. Patent 6,098,169).

12. As to claims 14 and 20, Milot teaches the invention substantially as claimed including: a computer-readable recording medium for recording a program to run on a computer, the program being operable to run on a computer, the program being operable to perform a simulation method of assigning threads of a OS to task processing threads of real-time OS to be simulated, and simulating an operation of said real-time OS on said OS (col. 4, lines 58-62), said simulation method comprising:

Receiving a request issued from a thread of the OS which is assigned as a processing thread (col. 4, lines 61-65; col. 5, lines 13-16; col. 6, lines 47-51);

Making a selected thread of the OS which is assigned as a task processing thread run by resuming the selected task processing thread according to the capabilities of OS (col. 5, lines 30-35).

13. Milot does not explicitly teach that providing an instruction to another thread of the OS which is assigned as task switching thread for switching the tasks in response to the request received in said receiving of the request. However, Ranganathan teaches providing an instruction to another thread of the OS which is assigned as task switching thread for switching the tasks in response to the request received in said receiving of the request (col. 5, lines 15-19; col. 6, lines 5-12).

14. It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of Milot and Ranganathan because Ranganathan's providing an instruction to another thread of the OS which is assigned as task switching thread for switching the tasks in response to the request received in said receiving of the request would optimize Milot et al's system by allowing another thread to easily control the switching of tasks.

15. As to claims 15 and 21, Milot teaches system function is operable to select a task processing thread to run next, to provide the instruction to said task switching thread for switching the currently running first task processing thread to

the thread of the OS which is assigned as the selected task processing thread, and to then suspend the first task processing thread (col. 2, lines 42-45; col. 4, lines 58-61; col. 5, lines 13-18, and lines 30-35), and

Said task switching thread is operable to, in response to the instruction provided by said system function, resume the selected task processing thread after the first task processing thread is suspended (col. 2, lines 52-60; col. 5, lines 30-35).

16. As to claims 16 and 22, Milot teaches system function is operable to select a task processing thread to run next, to provide the instruction to said task switching thread for switching the currently running task to the selected task processing thread (col. 6, lines 48-51), and to then set the thread of the OS corresponding to the first task processing thread that has issued the request in a waiting state (col. 5, lines 17-19), and

Said task switching thread is operable to, in response to the instruction, suspend the thread of the OS corresponding to the task (col. 5, lines 13-18), and then to release the selected task processing thread from a previously set waiting state for resuming (col. 5, lines 30-35).

17. As to claims 17 and 23, Milot teaches in said of the instruction, a task processing thread to run next is selected (col. 6, lines 47-51), and



Suspending the task processing thread issuing the request and resuming the selected task processing thread is given a higher priority than running the task processing threads, and, in said making of the selected task processing thread run, after the task processing thread issuing the request is suspended, the selected task processing thread is resumed. (Col. 5, lines 13-18 and lines 30-35).

18. As to claim 26, Milot teaches system function is operable to provide the instruction to said task switching thread after said task switching thread has been indicated as being enabled to start processing (col.5, lines 14-16).

19. Claims 8-9, and 27 are rejected under 35 U.S.C. 103(a) as being unpatentable over Milot et al (U.S. patent 6,437,788 B1), as applied to claim 1 above, in view of Hobbs et al (U.S. Patent 5,197,138).

20. As to claim 8, Milot teaches task switching thread is operable to select a thread to run next from among said task processing threads (col. 5, lines 30-35; col. 6, lines 48-52);

21. Milot does not explicitly teach that one of the pluralities of threads of the OS is assigned as an exception handling thread for task exception handling of each of the tasks. However, Hobbs teaches one of the pluralities of threads of

the OS is assigned as an exception handling thread for task exception handling of each of the tasks (col. 5, lines 31-33).

22. It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of Milot's and Hobbs because Hobbs's exception handling thread would improve the throughput of Milot's system by avoiding the problem of delayed exception.

23. As to claims 9 and 27, Milot teaches thread creating means for creating said task processing threads (col. 4, lines 58-61; col. 6, line 7 and line 26);

Hobbs teaches creating exception handling thread (col. 5, lines 31-33).

24. Claims 10-13, and 28-31 are rejected under 35 U.S.C. 103(a) as being unpatentable over Milot et al (U.S. patent 6,437,788 B1), as applied to claim 1 above, in view of Slaughter et al (U.S. Patent 6,738,846 B1).

25. As to claims 10-11, 28-29, Milot does not explicitly teach interrupt handling means for receiving an interrupt request issued by an interrupt thread that generate a pseudo-interrupt, suspending a running task processing thread, calling an interrupt handler corresponding to the received interrupt request, and then selecting a task processing thread to run next for resume. However, Slaughter teaches interrupt handling means for receiving an interrupt request

issued by an interrupt thread that generate a pseudo-interrupt (col. 5, lines 9-15), suspending a running task processing thread (col. 5, lines 15-18), calling an interrupt handler corresponding to the received interrupt request, and then selecting a task processing thread to run next for resume (col. 5, lines 15-25 and lines 30-37).

26. It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of Milot and Slaughter because Slaughter's interrupt handling would improve the mechanisms and frameworks of Milot's system by allowing cooperative execution of tasks using multiple threads in a multithread computer system.

27. As to claims 12 and 30, Slaughter teaches the interrupt thread includes a system clock interrupt thread that generates a pseudo-interrupt at predetermined time interval (col. 4, lines 20-39).

28. As to claims 13 and 31, Slaughter teaches interrupt thread means for creating the interrupt thread (col. 4, lines 20-23).

29. Claims 18 and 24 are rejected under 35 U.S.C. 103(a) as being unpatentable over Milot et al (U.S. Patent 6,437,788 B1) in view of Ranganathan

(U.S. Patent 6, 098, 169), as applied to claims 14 and 20 above, and further in view of Hobbs et al (U.S. Patent 5,197,138).

30. As to claims 18 and 24, Milot teaches making of the selected task processing thread run, wherein a thread to run next is selected from among said task processing threads (col. 5, lines 30-35; col. 6, lines 48-52).

31. Milot and Ranganathan do not explicitly teach that one of the pluralities of threads of the OS is assigned as an exception handling thread for task exception handling of each of the tasks. However, Hobbs teaches one of the pluralities of threads of the OS is assigned as an exception handling thread for task exception handling of each of the tasks (col. 5, lines 31-33).

32. It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teaching of Milot, Ranganathan and Hobbs because Hobbs's exception handling thread would improve the throughput of Milot and Ranganathan system by avoiding the problem of delayed exception.

33. Claims 19 and 25 are rejected under 35 U.S.C. 103(a) as being unpatentable over Milot et al (U.S. patent 6,437,788 B1) in view of Ranganathan (U.S. Patent 6,098,169), as applied to claims 14 and 20 above, and further in view of Slaughter et al. (U.S. Patent 6, 738, 846 B1).

34. As to claims 19 and 25, Milot and Ranganathan does not explicitly teach interrupt handling means for receiving an interrupt request issued by an interrupt thread that generate a pseudo-interrupt, suspending a running task processing thread, calling an interrupt handler corresponding to the received interrupt request, and then selecting a task processing thread to run next for resume.

35. However, Slaughter teaches interrupt handling means for receiving an interrupt request issued by an interrupt thread that generate a pseudo-interrupt (col. 5, lines 9-15), suspending a running task processing thread (col. 5, lines 15-18), calling an interrupt handler corresponding to the received interrupt request, and then selecting a task processing thread to run next for resume (col. 5, lines 15-25 and lines 30-37).

36. It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of Milot, Ranganathan and Slaughter because Slaughter's interrupt handling means would improve the mechanisms and frameworks of Milot's and Ranganathan's system by allowing cooperative execution of tasks using multiple threads in a multithread computer system.

### **Response to the argument**

37. Applicant's arguments filed 11/9/2004 for claims 1-31 have been considered but are moot in view of the new ground(s) rejection.

Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

### ***Conclusion***

38. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Camquy Truong whose telephone number is (571) 272-3773. The examiner can normally be reached on 8AM – 5PM.

Application/Control Number: 09/837,205  
Art Unit 2127

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Meng-Ai An can be reached on 571-272-3756. The fax phone number for the organization where this application or proceeding is assigned is 571-273-3756.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIP. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIP system, contact the Electronic Business Center (EBC) at 866-217-9197(toll-free).

Camquy Truong

February 17, 2005

  
MENG-AI T. AN  
SUPERVISORY PATENT EXAMINER  
TECHNOLOGY CENTER 2100